NOAA

National Environmental Satellite, Data, and Information Service (NESDIS)

Comprehensive Large Array-data Stewardship System (CLASS)

Office of Satellite Data Processing and Distribution (OSDPD) Activity Plan

(Version 1.0)

September 30, 2002

Revisions

Version	Description of Version	Date Completed
Draft 0.1	Initial draft	4/19/02
1.0	Incorporated CPMT comments – approved by CPMT	9/30/02

Review & Approval

Project Plan Review History

Reviewer	Version Reviewed	Signature	Date
Constantino Cremidis/CSC			
James Boney/CSC			
Alexander Kidd/OSDPD			
Geof Goodrum/NCDC			
Carlos Martinez/TMC			
Ted Habermann/NGDC			
Eric Kihn/NGDC			
David Vercelli/NESDIS			
Robert Mairs/NESDIS			

Contents

1 Project Definition

This document describes CSC's plan for work to be performed for the Office of Satellite Data Processing and Distribution (OSDPD), on the Comprehensive Large Array-data Stewardship System (CLASS) project over approximately the next 18 months (May 2002 to October 2003), referred to here as CLASS2003. It will be updated approximately twice per year to reflect progress and continuing plans for CLASS.

In addition to the CSC effort for CLASS, work is scheduled to be performed during this period at the National Climatic Data Center (NCDC) by TMC, and at the National Geophysical Data Center (NGDC). Where there are dependencies between CSC's planned work and that of another organization, those dependencies are identified in this document. This document does not explicitly include the scope or approach for the work of NCDC/TMC or NGDC.

Section 1 describes the overall scope and objectives for the CLASS2003 project. It includes the following topics:

- Overview – describes the project objectives and background
- Scope – defines the work included in the CLASS2003 implementation
- Deliverables – identifies work products that will be delivered to NOAA during this stage of implementation, and an overall timeline for their delivery
- Assumptions and Constraints – identifies the assumptions and constraints that underlie this plan. Changes to the assumptions or constraints will affect the scope, cost, and/or schedule included with this plan.
- Success and Completion Criteria – identifies the acceptance criteria for the project and the critical success factors
- Reference Material – refers the reader to additional material for more information on the CLASS project
- Definitions and Acronyms – provides definitions of terms and acronyms used in this document

1.1 Overview

The CLASS project is being conducted in support of the mission of the National Environmental Satellite, Data, and Information Service (NESDIS) to acquire, archive, and disseminate environmental data. NESDIS has been acquiring this data for more than 30 years, from a variety of *in situ* and remote sensing observing systems from throughout the National Oceanic and Atmospheric Administration (NOAA) and from a number of its partners. NESDIS foresees significant growth in both the data volume and the user population for this data, and has therefore initiated this effort to evolve current technologies to meet future needs.

The long-term goal for CLASS is the stewardship of all environmental data archived at the NOAA National Data Centers (NNDC). The initial objective for CLASS is to support specifically the following campaigns:

- NOAA and Department of Defense (DoD) Polar-orbiting Operational Environmental Satellites (POES)
- NOAA Geostationary-orbiting Operational Environmental Satellites (GOES)
- National Polar-orbiting Operational Environmental Satellite System (NPOESS)
- The NPOESS Preparatory Program (NPP)
- National Aeronautics and Space Administration (NASA) Earth Observing System (EOS)
- NOAA NEXt generation weather RADAR (NEXRAD) Program
- European Meteorological Operational Satellite (Metop) Program

The development of CLASS is expected to be a long-term, evolutionary process, as current and new campaigns are incorporated into the CLASS architecture. This project plan addresses approximately an 18-month period in CLASS development, the first stage in implementation of the CLASS architecture. During this period, the project will concentrate primarily on providing an overall framework for CLASS, and support for the NPP, GOES, EOS, and Metop campaigns.

1.2 Scope

1.2.1 Process

While the full CLASS architecture has not yet been completed, one goal for CLASS is to reuse existing systems where possible. The archive and distribution functionality for CLASS (ADS) is expected to be based primarily on the existing Satellite Active Archive (SAA) system. CSC's role over the next 18 months will include evolving the SAA system to a more generic CLASS ADS, while also supporting the development of the full system architecture and requirements for CLASS. Specifically, CSC is responsible for

- Enhancements of the SAA to facilitate incorporation of new campaigns, including
    - E-Commerce Interface – Interface with COAST in order to charge and receive payment for data
    - Delivery Process – Capability to deliver data via physical media
    - User Interface – Unified web interface, including replacement of the map server
    - Failover – SAA system setup at NCDC in Asheville, NC, to provide distribution of operations at two sites and failover capabilities
- Introduction of EOS, NPP, and Metop campaign support into CLASS ADS (EOS support for Moderate-Resolution Imaging Spectrometer (MODIS) to be operational

in CLASS2003; NPP and Metop campaigns support will be started, with support for end-to-end testing available in 2004.)

- Integration and test of the GOES campaign support implemented by NCDC/TMC
- Integration of the ADS component with other CLASS components, including
    - Development of the CLASS system requirements and architecture
    - Analysis of ADS subsystem requirements, based on the CLASS system requirements and architecture

Additionally, CSC will support

- The management team for the overall CLASS project.
- Data Discovery (DDS) activities performed by NGDC
- GOES development activities performed by NCDC/TMC
- DMSP/SABR functionality integration activities performed by NGDC.

1.2.2 Location/Organization

CSC's work on the CLASS2003 project will be performed at CSC and NOAA facilities in the Washington, DC area.

The operational system will be located at the NOAA facility in Suitland, MD, and the NCDC facility in Asheville, NC.

CSC will coordinate work with the NGDC CLASS teams in Boulder, CO, and the NCDC team (with TMC) in Asheville, NC and Fairmont, WV.

1.2.3 Data

Data for the following campaigns will be included in CLASS2003:

- EOS (MODIS)
- NPP (requirements, design, and initial development)
- Metop (requirements, design, and initial development)
- Data for the GOES campaign will be included in CLASS2003, however, that functionality is being implemented by the NCDC/TMC CLASS team. CSC will support TMC in familiarization with the SAA design and standards for development of the GOES functionality. CSC will also conduct integration and system test of the GOES functionality when it is incorporated into CLASS.

Data from the following campaigns is currently managed by SAA and will be inherited by CLASS2003:

- POES
- Existing derived products

1.2.4 Application

The following applications will be integrated into CLASS2003:

- SAA
- NOAA Coast watch Active Access System (NCAAS)
- Live Access Server (LAS) (current products website)
- Existing design for GOES Active Archive (GAA)

1.2.5 Technology

The following technology will be evaluated for the CLASS2003 implementation:

- Hardware and system software platform for the Suitland, MD operations center
- Programming Language: SAA is implemented primarily in C++. Evaluate appropriate use of JAVA.

The following technology components are defined for CLASS2003:

- Operating System: AIX, upgrade from version 4.3.3 to version 5.1
- Database: Informix, no change planned for CLASS2003

1.3 Deliverables

The following table identifies the major deliverables from CSC for CLASS2003, and the estimated delivery date. Delivery dates are based on the assumptions and constraints identified in Section 1.4, and are subject to change as those assumptions and constraints are modified.

Task	Deliverable	Expected Delivery Date
System Architecture	CLASS Architecture	September 2002
	CLASS/ADS SRR – consolidated CLASS Phase 1, POES, and GOES requirements and concept of operations	June 2002
System Infrastructure	Upgrade HW and SW at MD location	October 2002
	Setup distributed development CM server	June 2002 (Completed)
	Setup centralized integration and test environment	November 2002
	Operational system replicated at NCDC	April 2003
Implementation and Test:		
	CLASS/ADS v0.0	June 2002 (Completed)
	CLASS/ADS v1.0	December 2002

Integration of EOS campaign	Requirements Analysis	December 2002
	Preliminary and Detailed Design	April 2003
	Implementation and Deployment	October 2003
Integration of NPP campaign	Requirements Analysis	May 2003
	Preliminary and Detailed Design	September 2003
Integration of Metop campaign	Requirements Analysis	October 2002
	Preliminary Design	February 2003
	Detailed Design	June 2003

1.4 Assumptions and Constraints

The following assumptions and constraints underlie the project plan for CLASS2003. Changes to these factors could affect the cost, schedule, or scope of the work included in this plan.

- The communication link between DC and Asheville is upgraded and meets the needs for data transfer between the sites.
- Resources are allocated at Asheville to support knowledge transfer and for coordination of activities for replication of ADS operations at NCDC according to the above delivery schedule.
- There is an organization at Asheville prepared to support the system operations according to the above deliverable schedule.
- CSC has allocated staff to support CLASS activities that are being led by other team organizations. This includes the Communication Study, GOES IOC development and implementation, and requirements and concept of operations for the DDS.
    - The Communication Study is being conducted by NOAA, and is expected to be complete by December 2002. CLASS2003 is dependent on the results of that study for implementation of the dual site ADS operations and failover.
    - NCDC/TMC is responsible for the development of the GOES IOC and the implementation of GOES campaign support in CLASS. CSC will provide support to TMC in understanding the SAA design and the CLASS development standards and processes. CSC will also conduct integration and testing of the GOES functionality when it is ready for integration into CLASS. GOES IOC is expected to be complete by December 2002. GOES is expected to be ready for integration into CLASS by April 2003.
    - CSC expects the requirements and concept of operations for DDS to be complete by October 2002. The ADS requirements, concept of operations, and design will be revisited at that time to assure consistency with the DDS.

- Work on the NOAA E-Commerce Business Area Architecture (BAA) was completed. There is expected to be an interface between the E-Commerce component, or Order Management System (OMS), and CLASS ADS. This plan assumes that the OMS is defined by June 2002, so that CSC can begin work on the Interface Control Definition (ICD) in July 2002.
- CLASS2003 is dependent on the government for completion of the selection and acquisition of hardware for the Suitland facility. This plan assumes that the selection is complete by June 30, 2002 (completed; installation in progress).

1.5 Success and Completion Criteria

1.5.1 Acceptance Criteria

The acceptance test team will conduct acceptance testing for new releases of CLASS ADS at both operational sites, to ensure that both sites can perform the following functions in accordance with requirements:

- Ingest and archive data
- Search catalog
- Deliver data

Acceptance test criteria will be based on the system test plans developed by the CSC development team.

Formal acceptance of all CSC deliverables is determined by review and approval by the government CLASS Technical Lead, Alex Kidd.

1.5.2 Critical Success Factors

- Each team has a project plan that includes dependencies between teams, and provides sufficient intermediate milestones for tracking.
- Project progress is tracked to provide the CLASS Project Management Team (CPMT) with accurate and current status and forecasts.
- Risks are identified and managed by the CPMT.
- Standard development processes are defined and followed by all CLASS development teams.

1.6 Reference Materials

Documentation for the SAA is available online in the SAA library: library.saa.noaa.gov

1.7 Definitions and Acronyms

ADS	Archive and Distribution Subsystem
BAA	Business Area Architecture
CDF	Central Development Facility
CIO	Chief Information Officer
CLASS	Comprehensive Large Array-data Stewardship System
CM	Configuration Management
CMM	Capability Maturity Model
ComMIT	Comprehensive Management Information Tool
CPMT	CLASS Project Management Team
CSC	Computer Sciences Corporation
CSDPC	Central Satellite Data Processing Center
CVS	Concurrent Versions System
DDS	Data Discovery Subsystem
DoD	Department of Defense
EOS	Earth Observing System
GAA	GOES Active Archive
ICD	Interface Control Definition
IOC	Initial Operating Capability
IPD	Information Processing Division
LAS	Live Access Server
Metop	European Meteorological Operational Satellite Program
MODIS	Moderate-Resolution Imaging Spectrometer
NASA	National Aeronautics and Space Administration
NCAAS	NOAA Coast watch Active Access System
NCDC	National Climatic Data Center
NESDIS	National Environmental Satellite, Data, and Information Service
NEXRAD	NOAA NEXt generation weather RADAR Program
NGDC	National Geophysical Data Center
NNDC	NOAA National Data Centers
NOAA	National Oceanic and Atmospheric Administration
NODC	National Oceanic Data Center

NPOESS	National Polar-orbiting Operational Environmental Satellite System
NPP	NPOESS Preparatory Program
OMS	Order Management System
OSD	Office of System Development
OSDPD	Office of Satellite Data Processing and Distribution
PM	Project Manager
POES	NOAA and DoD Polar-orbiting Operational Environmental Satellites
QA	Quality Assurance
SAA	Satellite Active Archive
SEI	Software Engineering Institute
TBW	To be written

2 Management Structure

This section addresses the overall management approach CSC will use to manage the CLASS2003 effort.

2.1 Project Lifecycle

The CLASS2003 effort involves both maintenance/enhancement of the existing SAA system to support new campaigns, and development of the architecture for integration of SAA into CLASS. Hence, no single "lifecycle" applies to the full scope of the effort. The methodology for supporting these two concurrent activities will be based on CSC's Catalyst methodology, which provides a framework to support a variety of lifecycles. Additionally, CSC will use the guidance of the Software Engineering Institute (SEI) Software Capability Maturity Model (CMM) to drive key processes in the development effort.

The CSC approach will encompass the following major development phases:

Requirements Definition/Analysis: CSC will assess the new requirements for CLASS ADS based on the scheduled campaigns, as well as the new requirements derived from the CLASS system architecture and requirements. CSC will use the requirements management tool DOORS to document and track requirements, whatever their source. All requirements will be reviewed and approved by the CPMT before being scheduled for implementation. The requirements management process is based on the IPD Requirements Management Procedures (June 2000).

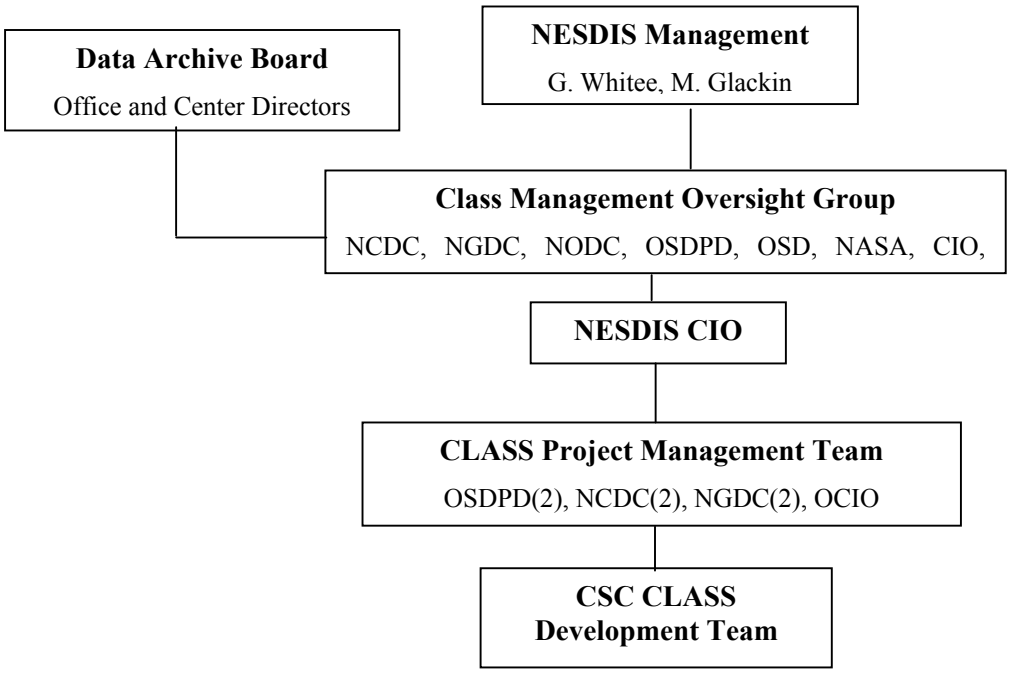
Preliminary/Detailed Design: CSC will complete the detailed design for each new capability planned for implementation in ADS before the start of that implementation. The design will undergo technical review before it is baselined for implementation. The overall design of the CLASS ADS component will be assessed and reviewed at the completion of the CLASS system architecture and requirements effort to identify any discrepancies between the SAA architecture and the CLASS ADS requirements.

Software Code and Test: CSC will implement the approved design following the CLASS coding standards and review processes. Individual developers will conduct initial unit and module testing, and an independent system test team will conduct full integration and system test of each release before promotion to operations. Coding standards are defined in the Software Standards for IPD (June 30, 2001).

Configuration Management: Software version control will be managed using the CVS tool, according to the processes defined in the CLASS Configuration Management Plan (October 1, 2002).

2.2 Project Organization

CSC is one member of a joint project team that also includes teams at NCDC and NGDC. The CLASS Project Management Team (CPMT) shown in the figure below manages this joint team.

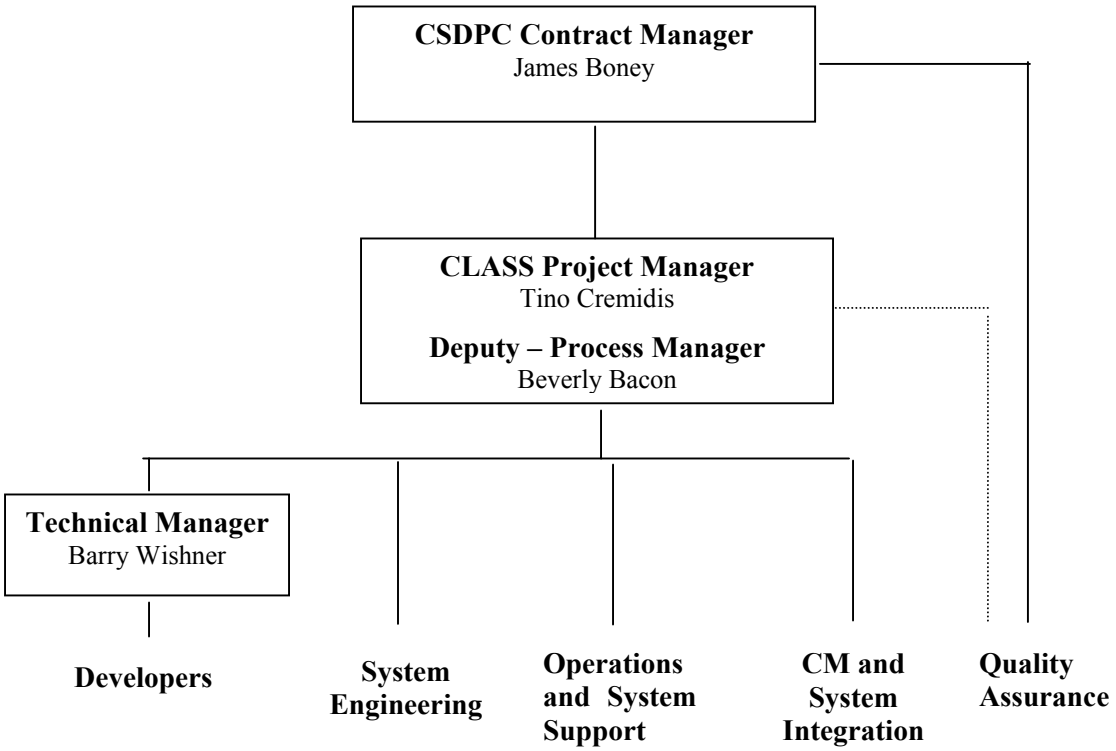


2.2.1 External Interfaces

Through the CPMT, the CSC CLASS project manager coordinates all CSC CLASS-related activities with the other organizations participating in CLASS development. The CPMT also provides the point of contact with other stakeholders, such as the current campaign operations teams for those campaigns planned for integration with CLASS (e.g., NASA).

2.2.2 Internal Structure

The following diagram shows the internal organization for the CSC CLASS team, and the key management personnel:



2.2.3 Roles and Responsibilities

Role	Responsibility
CSDPC Contract Manager	Oversees project performance and compliance with contract parameters; provides additional resources as required.
CLASS Project Manager	Leads CSC Development and Operations Teams; plans and monitors project activities; represents CSC Development Team on CPMT and serves as liaison between CSC team and the client
Deputy – Process Manager	Provides leadership and expertise in the definition, documentation, and application of processes for the full life-cycle; supports CLASS Project Manager in project planning and monitoring activities

Role	Responsibility
Technical Manager	Provides technical leadership for the CSC Development Team in design and implementation of the system; works with Project Manager to define detailed plans and identify required resources
Developers	Perform requirements analysis, design, code, and development testing.
System Engineering	Participates in CLASS system-level requirements, operations concept, and architecture activities; provides system expertise to the development team.
Operations and System Support	Provide support to the operational system in Suitland, MD
CM and System Integration	Conducts system-level configuration management, integration, and test for CLASS
Quality Assurance	Monitors project activities for conformance with project standards and processes; provides guidance to the project team on standards and processes

2.3 Communication

The following communication within the CSC CLASS team is necessary to ensure understanding and to provide appropriate visibility within the team:

- Weekly CSC CLASS development team meetings to review plans, status, issues, and risks
- Weekly written status report from each CSC team member to the CSC Project Manager (PM), and from the CSC PM to the CSDPC Contract Manager to report status and issues
- Monthly status report from the PM to the CSDPC Contract Manager to report performance against planned activities and budget

The following communication within the joint CLASS team is recommended to ensure appropriate coordination of the various organization development teams:

- Weekly CPMT teleconferences to review plans, status, issues, and risks
- Written project schedules from each development team
- Monthly progress reports to include task status and budget information (see Section 3.4 below)
- Monthly video-teleconference meeting of CPMT to review outstanding issues and risks
- Periodic meeting of full joint development teams to review status, process, and issues
- Joint reviews at the end of each phase: Requirements Definition/Analysis, Design, and Acceptance Test

2.4 Risk Management

CSC will maintain a list of risks that should be monitored or mitigated. For each risk identified, CSC will identify the probability of occurrence, the impact to the project, and a mitigation plan. Risks will be reviewed at the monthly CPMT meetings, and appropriate risk mitigation steps identified and approved.

3 Planning and Control

This section covers the planned approach for estimating, planning, and controlling the work.

3.1 Estimate

Estimates for CLASS2003 are primarily based on similar experience, i.e., actual effort and duration required for the previous SAA releases, and/or reported effort and duration for development of other existing systems that support the relevant campaigns. Where previous experience is not available (e.g., new technologies or programming language), the project will conduct prototypes before final estimation of the full system development effort.

3.2 Resource Identification

3.2.1 Staff

Upon approval of the CLASS2003 project plan, CSC will determine the staffing profile required to successfully complete the defined work. Staffing levels will be reviewed at each major milestone to determine if the available staff is adequate to support work planned.

3.2.2 Time

CLASS is expected to be an evolving system, as new campaigns are integrated into the base system. This plan addresses the next 18-month period only. Major milestones that will drive the activities scheduled for this period are the following:

- December 2002 – GOES Initial Operating Capability (IOC)
- December 2003 – EOS Archive and Distribution
- June 2004 – NPP end-to-end test
- October 2004 – Metop end-to-end test

While some of these milestone dates are outside the 18-month window for this plan, the activities included in this plan must be scheduled to meet the dates shown here.

3.2.3 Materials

All materials to be used on the CLASS2003 project will be supplied by NOAA.

The development equipment and integration and test system used by the CSC team will be resident at the NOAA Suitland facility.

3.3 Resource Allocation

3.3.1 Work Breakdown Structure

The following table shows the top two levels of the WBS for CLASS. Those activities for which the "Active Groups" list "All" or "CSC" are activities that are included in CSC's CLASS2003 plan. Lower levels are defined in the CLASS2003 MS Project schedule.

WBS #	Name	Description	Active Groups
1	Management		
1.1	Project Management		All
1.2	Quality Management		CSC
1.3	Configuration Management		CSC
1.4	Project Control		CSC
2	System Engineering		
2.1	CLASS System Architecture	Define, document, and maintain CLASS system architecture	All
2.2	CLASS System Requirements and Concept of Operations	Define, document, and maintain system requirements and concept of operations	All
2.3	CLASS Allocated Requirements	Define, document, and maintain allocated requirements	All
2.4	Interfaces	Define interfaces	All
2.5	Usability Assessment		NGDC-T
2.6	Engineering Research Grants		NGDC-T
3	System Infrastructure		
3.1	Upgrade Suitland Infrastructure	Assess and upgrade system hardware and software at Suitland facility	CSC
3.2	Replicate system at NCDC	Replicate operational environment at Asheville, NC facility	CSC, NCDC, TMC
3.3	NGDC Dev system		NGDC-E
3.4	COAST hardware & sys s/w		COAST
3.5	Tool support		CSC, NGDC-E, COAST
4	Implementation		
4.1	Implement CLASS R1 (Dec02)	Define requirements, design, and implement CLASS v1	CSC, TMC
4.2	CLASS R2 (June03)		CSC, TMC, NGDC-T
4.3	CLASS R3 (Dec03)		All
4.4	CLASS R4		CSC, TBD
5	Integration, Test, & Deployment		
5.1	CLASS R1		CSC, NCDC

5.2	CLASS R2		CSC, NCDC
5.3	CLASS R3		CSC, NCDC
5.4	User Training		COAST
6	Documentation		
6.1	CM Plan	Maintain CM documentation	CSC
6.2	QA Plan	Maintain QA documentation	CSC
6.3	Software Development Guide	Maintain SD Guide	CSC
6.4	Test Plan	Write and maintain CLASS Test Plan	CSC
6.5	Operations Support Plan	Write and maintain Op. Support Plan	CSC
7	CLASS Support		
7.1	Unplanned project support	Support each other as needed	All
7.2	System support		All
8	Operations Support	Operational support for Suitland operations	
8.1	Asheville Ops		NCDC, TMC
8.2	Suitland Ops		CSC
8.3	Science support		All

3.3.2 Schedule

The detailed schedule for CLASS2003 is provided in the MS Project file, CLASS2003.mpp. The high-level milestones for deliverables are provided in Section 1.3.

3.4 Tracking and Control

CLASS2003 cost and schedule will be managed using Earned Value Measurement techniques, as described in the CLASS Master Project Management Plan. Task status will be maintained in the CLASS2003 MS Project file and input to Comprehensive Management Information Tool (ComMIT), and actual cost data will be input to ComMIT from CSC's CostPoint tool. ComMIT can then be used to generate a series of project performance reports to include cost, budget, and schedule variances, trends, and estimates to complete. These reports will be generated monthly to provide CSC and CLASS management with the necessary performance information to successfully manage the CLASS project.

Quality of the products produced for CLASS2003 will be monitored through peer review, independent testing, and periodic process audits, as defined in the CLASS Quality Management Plan (August 30, 2002) and the CLASS Software Development Guide (July 22, 2002).

Functionality of the CLASS system will be tracked using the DOORS tool for requirements tracking, and CVS for software version control, as described in the CLASS Configuration Management Plan (October 1, 2002).

4 Technical Process

This section summarizes the top-level technical processes used on this project. Details about the technical approaches will be expanded in the Software Development Guide

4.1 Engineering

CSC will provide system and software engineering processes in the Software Development Guide for use throughout the CLASS project for all phases of the development life cycle. Systems engineering is the discipline for establishing a sound system concept, defining and validating clear and concise system requirements, developing candidate solutions for an effective system design, and ensuring that the developed system meets client and user objectives and needs in the operational environment. Software engineering further describes the discipline around the detailed design, code, and test of the allocated functionality, following proven processes and employing appropriate tools to ensure the successful development, deployment, maintenance, and operation of CLASS.

4.1.1 Environment

A CLASS Systems Engineering Team (SET), following the processes defined in the Software Development Guide, will coordinate the engineering functions. This will be a multi-disciplinary team consisting of the CSC Technical Manager and other staff members from CSC, NCDC, and NGDC. This cross-organizational team will assure that all CLASS system-level requirements are defined and allocated to the appropriate organization for analysis and implementation.

4.1.2 Methods, Tools and Techniques

The activities of the SET will include:

- System definition and design:
    - Validate a comprehensive system concept
    - Validate system-level requirements derived from approved system concept
    - Validate the system architecture and allocation of system requirements to subsystems
    - Document and maintain system requirements and their allocation throughout the system life cycle
    - Develop system performance and system effectiveness metrics
    - Define system acceptance criteria
- Implementation:
    - Ensure compliance of detailed design with system-level requirements

  - Conduct requirement-change impact assessment
  - Provide workaround and fallback solutions to critical design issues
  - Measure and evaluate system effectiveness
- Integration and test:
  - Verify compliance of test definition and test results with system-level requirements
  - Resolve technical issues
  - Support transition to operations

After the requirements have been allocated to a specific development team, the development team activities will include:

- Software and database design
- Software implementation
- Testing
- Configuration Management
- Documentation
- Problem reporting and tracking

CSC will maintain central CM and system test teams to manage the integration and delivery of the system releases.

The CLASS Software Development Guide will describe the processes to be followed in each of these activities. CSC will use the requirements management tool DOORS to document requirements and to maintain traceability from requirements to design elements and to test cases.

4.2 Technology

This section describes the physical development environment and the tools and methods used to support development.

4.2.1 Environment

CLASS software will be developed and maintained at several development sites linked to a Central Development Facility (CDF), located in Suitland, MD. The development sites will be in:

- Suitland, MD
- Asheville, NC,
- Fairmont, WV
- Boulder, CO.

CSC will setup and maintain the software repository server at the CDF. It is the responsibility of each site to setup and maintain their own local development environment. CSC will provide assistance as needed to setup the local development environment. Developers at each site may check out software, copy it to their local systems, develop and test software locally, and check in finished products at the CDF.

CSC's configuration management team will be responsible for promoting software to the integration and test environments and to operations. Each release will be integrated and independently tested at the CDF, and distributed from there to the operational sites.

The operational system will be located at the NOAA facility in Suitland, MD, and the NCDC facility in Asheville, NC. These two sites will share a database and will archive and have access to the same data files so that failover can be effected quickly and either site can fill any order and support all operations. Test teams at each operational facility will conduct acceptance testing of each software release.

4.2.2 Methods, Tools, and Techniques

A variety of tools are currently in use to support technical activities. New tools may be added, upon SET approval, as CLASS evolves. The current tool set includes the following:

- The Telelogic package Tau-UML (formerly Object Team) supports object-oriented design with the Unified Modeling Language (UML). It generates various diagrams that are useful for analysis and design (e.g., Use Case, Activity, State, Sequence, Class Association diagrams), verifies design consistency, and generates reports.
- The primary programming languages used are C++, Java, JavaScript, and Perl.
- Off-the-shelf components are used in the Java/XML-based user interface:
    - Apache - HTTP server
    - Cocoon - publishing framework
    - Turbine - database connection pooling
    - Tomcat - servlet engine
    - Log4j - message logging
    - Informix Java Database Connectivity (JDBC) Driver
- Perl scripts are used to automate testing
- The freely available Concurrent Versions System (CVS) provides software version control in the distributed development environment.
- A suite of custom-built Perl programs is used to manage and log software promotions
- Off-the-shelf tools are use to generate on-line software reference documentation in HTML format (e.g., Javadoc, Doxygen)
- The Remedy Change Management system is used to creates and track problem reports

- The Informix database supports all phases of development, configuration management, and operations
- ERWin is used for database design.

5 Supporting Plans

This section references the detailed supporting plans.

5.1 Configuration Management

CLASS Configuration Management Plan, October 1, 2002

5.2 Quality Assurance

CLASS Quality Management Plan, draft, August 30, 2002

5.3 Testing

Regression and acceptance test plans, TBW

5.4 Software Development

Software Standards for Information Processing Division (IPD), June 30, 2001

CLASS Software Development Guide, draft, July 22, 2002

5.5 Operations Support

CLASS Operations Manual, TBW

CLASS System Support Guide, TBW